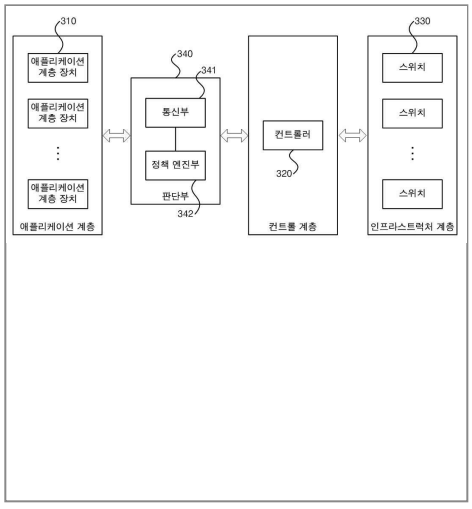
악성 애플리케이션을 방지할 수 있는 소프트웨어 정의 네트워크 및 이에 포함되는 판단 장치

SDN for preventing malicious application and Determination apparatus comprising the same

(51) Int. CL	H04L 29/06(2006.01.01)
(52) CPC	H04L 63/16(2013.01) H04L 63/20(2013.01) H04L 63/1441(2013.01)
(21) Application No.(Date)	1020160134426 (2016.10.17)
(71) Applicant	Foundation of Soongsil University-Industry Cooperation
(11) Registration No.(Date)	1018549960000 (2018.04.27)
(65) Unex. Pub. No.(Date)	1020180041976 (2018.04.25)
(11) Publication No.(Date)	(2018.05.04)
(86) Int'l Application No.(Date)	
(87) Int'l Unex. Pub. No.(Date)	
(30) Priority info. (Country / No. / Date)	
Legal Status	Registered
Examination Status	Decision to grant (General)
Trial Info	
Kind	Domestic Application / New Application
Right of Org. Application No.(Date)	
Related Application No.	
Request for an examination(Date)	Y(2016.10.17)
Number of examination claims	5



KPA (Korea Patent Abstract)

Disclosed are a software defined network capable of preventing a malicious application, and a determination device included therein. The disclosed software defined network including an application layer, a control layer, and an infrastructure layer, comprises: a controller located on the control layer and installed with an application programming interface (API) and an application, wherein the application is any one of a normal application and a malicious application; and a determination unit for receiving an API request for using the installed application from the application

layer, and determining whether the received API request is an API request for the normal application or for the malicious application. The determination unit transmits the received API request to the controller when the received API request is the API request for the normal application.

(71) Applicant

No.	Name	Country	Address
1	Foundation of Soongsil University-Industry Cooperation 숭실대학교산학협력단 (220060278499)	Korea	서울특별시 동작구 상도로 *** (상도동)

(72) Inventor

No.	Name	Country	Address
1	YOO, Myung Sik 유명식	Republic of Korea	서울특별시 서초구...
2	Nguyen Tri Hai 응웬트리하이	Viet Nam	서울특별시 동작구 상도로 ***, 형남공학관 ****호 (상도동)
3	CHOI, Jin Seok 최진석	Republic of Korea	서울특별시 동작구...

(74) Agent

No.	Name	Country	Address
1	Song in ho 송인호 (920050008309)	Korea	*th Floor Donglim Bldg. **, Gangnam-daero **-gil, Gangnam-gu, Seoul(IP-WIZ INTERNATIONAL PATENT & LAW OFFICE)
2	choi kwan rak 최관락 (920040002237)	Korea	*th Floor Donglim Bldg. **, Gangnam-daero **-gil, Gangnam-gu, Seoul(IP-WIZ INTERNATIONAL PATENT & LAW OFFICE)

Right holder(current)

Name	Country	Address
숭실대학교산학협력단	KR	서울특별시 동작구...

Legal Status

No.	Document Title(Eng.)	Receipt/Delivery Date	Status	Receipt/Delivery No.
1	[Patent Application] Patent Application ([특허출원]특허출원서)	2016.10.17	Accepted (수리)	112016100383455
2	Notification of reason for refusal (의견제출통지서)	2018.01.22	Completion of Transmission (발송처리완료)	952018005047405
3	[Amendment to Description, etc.] Amendment ([명세서등 보정]보정서)	2018.02.01	Regarded as an acceptance of amendment (보정승인간주)	112018011787557
4	[Opinion according to the Notification of Reasons for Refusal] Written Opinion(Written Reply, Written Substantiation) ([거절이유 등 통지에 따른 의견]의견(답변, 소명)서)	2018.02.01	Accepted (수리)	112018011787412
5	Decision to grant (등록결정서)	2018.04.20	Completion of Transmission (발송처리완료)	952018027451194

Machine Translation

Claim

No.	Content
1	애플리케이션 계층, 컨트롤 계층 및 인프라스트럭처 계층으로 구성되는 소프트웨어 정의 네트워크에 있어서, 상기 컨트롤 계층 상에 위치하며, API(Application Programming Interface) 및 애플리케이션이 설치되는 컨트롤러 – 상기 애플리케이션은 정상 애플리케이션 또는 악성 애플리케이션 중 어느 하나임 –; 상기 설치된 애플리케이션을 사용하기 위한 API 요청을 상기 애플리케이션 계층으로부터 수신하는 통신부와, 상기 정상 애플리케이션에 대한 API 요청인지 여부를 판단하는 정책 정보를 이용하여 상기 수신된 API 요청이 상기 정상 애플리케이션을 위한 API 요청인지 상기 악성 애플리케이션을 위한 API 요청인지 여부를 판단하는 정책 엔진부를 포함하는 판단부;를 포함하되, 상기 정책 정보는 기 설정된 범위에서 변화하는 동적 데이터의 정보인 제3 정책 정보를 포함하고, 상기 정책 엔진부는 상기 수신된 API 요청에 포함된 동적 데이터가 상기 제3 정책 정보에 포함된 동적 데이터의 정보에 포함되는 경우 상기 수신된 API 요청이 상기 정상 애플리케이션을 위한 API 요청인 것으로 판단하며, 상기 통신부는 상기 수신된 API 요청이 상기 정상 애플리케이션을 위한 API 요청인 경우 상기 수신된 API 요청을 상기 컨트롤러로 전송하는 것을 특징으로 하는 소프트웨어 정의 네트워크.
2	삭제

No.	Content
3	제1항에 있어서, 상기 정책 정보는 상기 API에 접근 가능한 적어도 하나의 애플리케이션의 식별정보인 제1 정책 정보를 포함하고, 상기 정책 엔진부는 상기 수신된 API 요청에 포함된 애플리케이션의 식별정보가 상기 제1 정책 정보에 포함된 적어도 하나의 애플리케이션의 식별정보 중 하나인 경우 상기 수신된 API 요청이 상기 정상 애플리케이션을 위한 API 요청인 것으로 판단하는 것을 특징으로 하는 소프트웨어 정의 네트워크.
4	제1항에 있어서, 상기 정책 정보는 상기 API로 전달될 수 있는 불변(invariant) 데이터의 정보인 제2 정책 정보를 포함하고, 상기 정책 엔진부는 상기 수신된 API 요청에 포함된 불변 데이터 정보가 상기 제2 정책 정보에 포함된 불변 데이터의 정보에 포함되는 경우 상기 수신된 API 요청이 상기 정상 애플리케이션을 위한 API 요청인 것으로 판단하는 것을 특징으로 하는 소프트웨어 정의 네트워크.
5	삭제
6	제1항에 있어서, 상기 정책 정보는 2개의 모듈 사이의 통신의 흐름 정보인 제4 정책 정보를 포함하고 – 상기 모듈은 상기 인프라스트럭처 계층에 존재하는 모듈임 –, 상기 정책 엔진부는 상기 수신된 API 요청에 포함된 통신의 흐름 정보가 상기 제4 정책 정보에 포함된 통신의 흐름 정보에 포함되는 경우 상기 수신된 API 요청이 상기 정상 애플리케이션을 위한 API 요청인 것으로 판단하는 것을 특징으로 하는 소프트웨어 정의 네트워크.
7	소프트웨어 정의 네트워크 내의 애플리케이션 계층과 컨트롤 계층 사이에 존재하는 악성 애플리케이션 판단 장치에 있어서, 상기 컨트롤 계층 상에 위치하는 컨트롤러에 설치된 애플리케이션을 사용하기 위한 API 요청을 상기 애플리케이션 계층으로부터 수신하는 통신부; 및 정상 애플리케이션에 대한 API 요청인지 여부를 판단하는 정책 정보를 이용하여 상기 수신된 API 요청이 상기 정상 애플리케이션을 위한 API 요청인지 상기 악성 애플리케이션을 위한 API 요청인지 여부를 판단하는 정책 엔진부;를 포함하되, 상기 정책 정보는 기 설정된 범위에서 변화하는 동적 데이터의 정보인 제3 정책 정보를 포함하고, 상기 정책 엔진부는 상기 수신된 API 요청에 포함된 동적 데이터가 상기 제3 정책 정보에 포함된 동적 데이터의 정보에 포함되는 경우 상기 수신된 API 요청이 상기 정상 애플리케이션을 위한 API 요청인 것으로 판단하는 것을 특징으로 하는 악성 애플리케이션 판단 장치.

Designated States

Kind	Country
:: Empty ::	

※ The information is based on the citation information attached to a Notification of reason for refusal by the examiner.

Forward Citation

Citation

Country	Pub. Date	Pub. No	Title	IPC
Republic of Korea	1020090096823 A	2009.09.15	-	-

Backward Citation

Citation

Application No	Application Date	Title	IPC
:: Empty ::			

Patent Kind Codes

View Graph

Family Patents

No.	Family No.	Country(code)	Country	Type
::Empty::				

DOCDB Family info.

Family Patents

No.	Family No.	Country(code)	Country	Type
::Empty::				

(19) 대한민국특허청(KR)
(12) 등록특허공보(B1)

(45) 공고일자 2018년05월04일
(11) 등록번호 10-1854996
(24) 등록일자 2018년04월27일

(51) 국제특허분류(Int. Cl.)
H04L 29/06 (2006.01)
(52) CPC특허분류
H04L 63/16 (2013.01)
H04L 63/1441 (2013.01)
(21) 출원번호 10-2016-0134426
(22) 출원일자 2016년10월17일
심사청구일자 2016년10월17일
(65) 공개번호 10-2018-0041976
(43) 공개일자 2018년04월25일
(56) 선행기술조사문헌
KR1020090096823 A*
KR1020070080354 A
KR1020130085483 A
*는 심사관에 의하여 인용된 문헌

(73) 특허권자
숭실대학교산학협력단
서울특별시 동작구 상도로 369 (상도동)
(72) 발명자
유명식
서울특별시 서초구 신반포로3길 19 , 96동 408호 (반포동)
웅웬트리하이
서울특별시 동작구 상도로 369, 형남공학관 1103호 (상도동)
최진석
서울특별시 동작구 등용로 37, 108동 1609호 (상도동, 상도래미안1차아파트)
(74) 대리인
송인호, 최관락

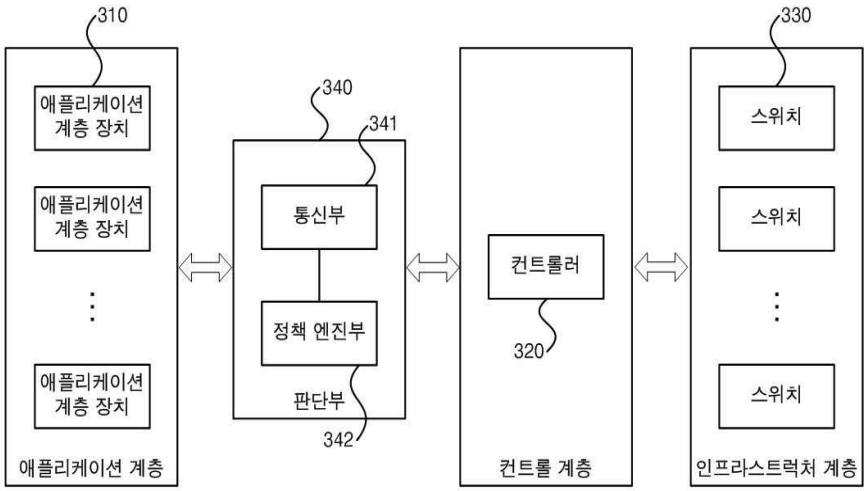
전체 청구항 수 : 총 5 항 심사관 : 남기영

(54) 발명의 명칭 **악성 애플리케이션을 방지할 수 있는 소프트웨어 정의 네트워크 및 이에 포함되는 판단 장치**

(57) 요 약

악성 애플리케이션을 방지할 수 있는 소프트웨어 정의 네트워크 및 이에 포함되는 판단 장치가 개시된다. 개시된 소프트웨어 정의 네트워크는 애플리케이션 계층, 컨트롤 계층 및 인프라스트럭처 계층으로 구성되되, 상기 컨트롤 계층 상에 위치하며, API(Application Programming Interface) 및 애플리케이션이 설치되는 컨트롤러 - 상기 애플리케이션은 정상 애플리케이션 또는 악성 애플리케이션 중 어느 하나임 -; 상기 설치된 애플리케이션을 사용하기 위한 API 요청을 상기 애플리케이션 계층으로부터 수신하고, 상기 수신된 API 요청이 상기 정상 애플리케이션을 위한 API 요청인지 상기 악성 애플리케이션을 위한 API 요청인지 여부를 판단하는 판단부;를 포함하되, 상기 판단부는 상기 수신된 API 요청이 상기 정상 애플리케이션을 위한 API 요청인 경우, 상기 수신된 API 요청을 상기 컨트롤러로 전송한다.

대 표 도 - 도3



(52) CPC특허분류
H04L 63/20 (2013.01)

이 발명을 지원한 국가연구개발사업
과제고유번호 H8501-16-1008 / 1711035246
부처명 미래창조과학부
연구관리전문기관 정보통신기술진흥센터
연구사업명 대학 ICT연구센터 육성 지원사업
연구과제명 클라우드 환경의 스마트 기기와 서비스 보안 기술 개발 및 연구 인력양성
기 여 율 1/1
주관기관 숭실대학교산학협력단
연구기간 2016.01.01 ~ 2016.12.31

명 세 서

청구범위

청구항 1

애플리케이션 계층, 컨트롤 계층 및 인프라스트럭처 계층으로 구성되는 소프트웨어 정의 네트워크에 있어서,

상기 컨트롤 계층 상에 위치하며, API(Application Programming Interface) 및 애플리케이션이 설치되는 컨트롤러 - 상기 애플리케이션은 정상 애플리케이션 또는 악성 애플리케이션 중 어느 하나임 -;

상기 설치된 애플리케이션을 사용하기 위한 API 요청을 상기 애플리케이션 계층으로부터 수신하는 통신부와, 상기 정상 애플리케이션에 대한 API 요청인지 여부를 판단하는 정책 정보를 이용하여 상기 수신된 API 요청이 상기 정상 애플리케이션을 위한 API 요청인지 상기 악성 애플리케이션을 위한 API 요청인지 여부를 판단하는 정책 엔진부를 포함하는 판단부;를 포함하되,

상기 정책 정보는 기 설정된 범위에서 변화하는 동적 데이터의 정보인 제3 정책 정보를 포함하고, 상기 정책 엔진부는 상기 수신된 API 요청에 포함된 동적 데이터가 상기 제3 정책 정보에 포함된 동적 데이터의 정보에 포함되는 경우 상기 수신된 API 요청이 상기 정상 애플리케이션을 위한 API 요청인 것으로 판단하며,

상기 통신부는 상기 수신된 API 요청이 상기 정상 애플리케이션을 위한 API 요청인 경우 상기 수신된 API 요청을 상기 컨트롤러로 전송하는 것을 특징으로 하는 소프트웨어 정의 네트워크.

청구항 2

삭제

청구항 3

제1항에 있어서,

상기 정책 정보는 상기 API에 접근 가능한 적어도 하나의 애플리케이션의 식별정보인 제1 정책 정보를 포함하고,

상기 정책 엔진부는 상기 수신된 API 요청에 포함된 애플리케이션의 식별정보가 상기 제1 정책 정보에 포함된 적어도 하나의 애플리케이션의 식별정보 중 하나인 경우 상기 수신된 API 요청이 상기 정상 애플리케이션을 위한 API 요청인 것으로 판단하는 것을 특징으로 하는 소프트웨어 정의 네트워크.

청구항 4

제1항에 있어서,

상기 정책 정보는 상기 API로 전달될 수 있는 불변(invariant) 데이터의 정보인 제2 정책 정보를 포함하고,

상기 정책 엔진부는 상기 수신된 API 요청에 포함된 불변 데이터 정보가 상기 제2 정책 정보에 포함된 불변 데이터의 정보에 포함되는 경우 상기 수신된 API 요청이 상기 정상 애플리케이션을 위한 API 요청인 것으로 판단하는 것을 특징으로 하는 소프트웨어 정의 네트워크.

청구항 5

삭제

청구항 6

제1항에 있어서,

상기 정책 정보는 2개의 모듈 사이의 통신의 흐름 정보인 제4 정책 정보를 포함하고 - 상기 모듈은 상기 인프라스트럭처 계층에 존재하는 모듈임 -,

상기 정책 엔진부는 상기 수신된 API 요청에 포함된 통신의 흐름 정보가 상기 제4 정책 정보에 포함된 통신의

흐름 정보에 포함되는 경우 상기 수신된 API 요청이 상기 정상 애플리케이션을 위한 API 요청인 것으로 판단하는 것을 특징으로 하는 소프트웨어 정의 네트워크.

청구항 7

소프트웨어 정의 네트워크 내의 애플리케이션 계층과 컨트롤 계층 사이에 존재하는 악성 애플리케이션 판단 장치에 있어서,

상기 컨트롤 계층 상에 위치하는 컨트롤러에 설치된 애플리케이션을 사용하기 위한 API 요청을 상기 애플리케이션 계층으로부터 수신하는 통신부; 및

정상 애플리케이션에 대한 API 요청인지 여부를 판단하는 정책 정보를 이용하여 상기 수신된 API 요청이 상기 정상 애플리케이션을 위한 API 요청인지 상기 악성 애플리케이션을 위한 API 요청인지 여부를 판단하는 정책 엔진부;를 포함하되,

상기 정책 정보는 기 설정된 범위에서 변화하는 동적 데이터의 정보인 제3 정책 정보를 포함하고,

상기 정책 엔진부는 상기 수신된 API 요청에 포함된 동적 데이터가 상기 제3 정책 정보에 포함된 동적 데이터의 정보에 포함되는 경우 상기 수신된 API 요청이 상기 정상 애플리케이션을 위한 API 요청인 것으로 판단하는 것을 특징으로 하는 악성 애플리케이션 판단 장치.

발명의 설명

기 술 분 야

[0001] 본 발명의 실시예들은 악성 애플리케이션을 방지할 수 있는 소프트웨어 정의 네트워크(SDN: Software Defined Network) 및 이에 포함되는 판단 장치에 관한 것이다.

배 경 기 술

[0002] 인터넷은 우리의 일상에서 이제 불가분의 중요한 역할을 하고 있으며 사물 인터넷이 본격적으로 일상에 적용될 시에는 이 역할은 더욱 커질 것이라 예상한다. 하지만 종래의 네트워크 장비는 미리 정해진 룰에 따라 작동이 되는 시스템으로서, 관리 시 어려움이 있으며, 새로운 기능을 추가 할 시에는 연관된 모든 장비를 업데이트 또는 교체해야 하는 불편함이 존재한다. 그리고, 각종 새로운 악성 공격으로부터도 보안 상의 취약성을 보이고 있다.

[0003] 따라서, 이를 해결하고자 등장한 것이 소프트웨어 정의 네트워크(SDN: Software Defined Network)이다. 소프트웨어 정의 네트워크는 기존의 네트워크 장비와는 달리 컨트롤 평면(control plane)과 데이터 평면(data plane)이 분리된다. 따라서, 네트워크 구조가 단순화되어 있고, 네트워크 관리를 유연하게 해주며, 기존 네트워크보다 악성 공격에 대하여 일부 강점이 있다. 하지만, 소프트웨어 정의 네트워크도 보안에 관하여는 완벽한 해결책이 없으며 여전히 취약한 면이 있는 것도 사실이다.

[0004] 그 중에서도 악성 애플리케이션 공격은 전체 네트워크를 강제적으로 종료하게 만들 수 있는 공격이다. 즉, 소프트웨어 정의 네트워크 내의 컨트롤 평면 상의 컨트롤러에 가해지는 악성 애플리케이션 공격은 공격자가 컨트롤러의 관리자로부터 악성 애플리케이션을 다운로드하도록 유도한 후, 컨트롤러의 권한을 빼앗아 쉽게 네트워크를 장악하거나 마비시킬 수 있으며, 심지어는 다른 정상 애플리케이션까지 삭제해버리는 공격까지 시도할 수 있다.

이러한 문제점을 해결하기 위한 관련 선행문헌으로서, 대한민국 공개특허 제10-2016-0002269호(발명명칭: SDN 기반의 ARP 스푸핑 탐지장치 및 그 방법)가 있다.

발명의 내용

해결하려는 과제

[0005] 상기한 바와 같은 종래기술의 문제점을 해결하기 위해, 본 발명에서는 소프트웨어 정의 네트워크의 컨트롤러의 API(Application Programming Interface)를 오용으로부터 보호하며, 악성 애플리케이션의 공격으로부터 컨트롤러를 보호하는 소프트웨어 정의 네트워크 및 이에 포함되는 판단 장치를 제공하고자 한다.

[0006] 본 발명의 다른 목적들은 하기의 실시예를 통해 당업자에 의해 도출될 수 있을 것이다.

과제의 해결 수단

[0007] 상기한 목적을 달성하기 위해 본 발명의 바람직한 일 실시예에 따르면, 애플리케이션 계층, 컨트롤 계층 및 인프라스트럭처 계층으로 구성되는 소프트웨어 정의 네트워크에 있어서, 상기 컨트롤 계층 상에 위치하며, API(Application Programming Interface) 및 애플리케이션이 설치되는 컨트롤러 - 상기 애플리케이션은 정상 애플리케이션 또는 악성 애플리케이션 중 어느 하나임 -; 상기 설치된 애플리케이션을 사용하기 위한 API 요청을 상기 애플리케이션 계층으로부터 수신하고, 상기 수신된 API 요청이 상기 정상 애플리케이션을 위한 API 요청인지 상기 악성 애플리케이션을 위한 API 요청인지 여부를 판단하는 판단부;를 포함하되, 상기 판단부는 상기 수신된 API 요청이 상기 정상 애플리케이션을 위한 API 요청인 경우, 상기 수신된 API 요청을 상기 컨트롤러로 전송하는 것을 특징으로 하는 소프트웨어 정의 네트워크가 제공된다.

[0008] 상기 판단부는, 상기 API 요청을 수신하고, 상기 수신된 API 요청이 상기 정상 애플리케이션인 경우 상기 수신된 API 요청을 상기 컨트롤러로 전송하는 통신부; 및 정상 애플리케이션에 대한 API 요청인지 여부를 판단하는 정책 정보를 이용하여 상기 수신된 API 요청이 상기 정상 애플리케이션을 위한 API 요청인지 상기 악성 애플리케이션을 위한 API 요청인지 여부를 판단하는 정책 엔진부;를 포함할 수 있다.

[0009] 상기 정책 정보는 상기 API에 접근 가능한 적어도 하나의 애플리케이션의 식별정보인 제1 정책 정보를 포함하고, 상기 정책 엔진부는 상기 수신된 API 요청에 포함된 애플리케이션의 식별정보가 상기 제1 정책 정보에 포함된 적어도 하나의 애플리케이션의 식별정보 중 하나인 경우 상기 수신된 API 요청이 상기 정상 애플리케이션을 위한 API 요청인 것으로 판단할 수 있다.

[0010] 상기 정책 정보는 상기 API로 전달될 수 있는 불변(invariant) 데이터의 정보인 제2 정책 정보를 포함하고, 상기 정책 엔진부는 상기 수신된 API 요청에 포함된 불변 데이터 정보가 상기 제2 정책 정보에 포함된 불변 데이터의 정보에 포함되는 경우 상기 수신된 API 요청이 상기 정상 애플리케이션을 위한 API 요청인 것으로 판단할 수 있다.

[0011] 상기 정책 정보는 기 설정된 범위에서 변화하는 동적 데이터의 정보인 제3 정책 정보를 포함하고, 상기 정책 엔진부는 상기 수신된 API 요청에 포함된 동적 데이터가 상기 제3 정책 정보에 포함된 동적 데이터의 정보에 포함되는 경우 상기 수신된 API 요청이 상기 정상 애플리케이션을 위한 API 요청인 것으로 판단할 수 있다.

[0012] 상기 정책 정보는 2개의 모듈 사이의 통신의 흐름 정보인 제4 정책 정보를 포함하고 - 상기 모듈은 상기 인프라스트럭처 계층에 존재하는 모듈임 -, 상기 정책 엔진부는 상기 수신된 API 요청에 포함된 통신의 흐름 정보가 상기 제4 정책 정보에 포함된 통신의 흐름 정보에 포함되는 경우 상기 수신된 API 요청이 상기 정상 애플리케이션을 위한 API 요청인 것으로 판단할 수 있다.

[0013] 또한, 본 발명의 다른 실시예에 따르면, 소프트웨어 정의 네트워크 내의 애플리케이션 계층과 컨트롤 계층 사이에 존재하는 악성 애플리케이션 판단 장치에 있어서, 상기 컨트롤 계층 상에 위치하는 컨트롤러에 설치된 애플리케이션을 사용하기 위한 API 요청을 상기 애플리케이션 계층으로부터 수신하는 통신부; 및 정상 애플리케이션에 대한 API 요청인지 여부를 판단하는 정책 정보를 이용하여 상기 수신된 API 요청이 상기 정상 애플리케이션을 위한 API 요청인지 상기 악성 애플리케이션을 위한 API 요청인지 여부를 판단하는 정책 엔진부;를 포함하는 것을 특징으로 하는 악성 애플리케이션 판단 장치가 제공된다.

발명의 효과

[0014] 본 발명에 따른 소프트웨어 정의 네트워크 및 이에 포함되는 판단 장치는 소프트웨어 정의 네트워크의 컨트롤러의 API를 오용으로부터 보호하며, 악성 애플리케이션의 공격으로부터 컨트롤러를 보호하는 장점이 있다.

도면의 간단한 설명

[0015] 도 1은 소프트웨어 정의 네트워크(SDN: Software Defined Network)의 기본 구조를 도시한 도면이다.

도 2는 소프트웨어 정의 네트워크에 사용되는 OpenFlow의 구조를 도시한 도면이다.

도 3은 본 발명의 일 실시예에 따른 소프트웨어 정의 네트워크(SDN: Software Defined Network)의 개략적인 구조를 도시한 도면이다.

발명을 실시하기 위한 구체적인 내용

[0016] 본 명세서에서 사용되는 단수의 표현은 문맥상 명백하게 다르게 뜻하지 않는 한, 복수의 표현을 포함한다. 본 명세서에서, "구성된다" 또는 "포함한다" 등의 용어는 명세서상에 기재된 여러 구성 요소들, 또는 여러 단계들을 반드시 모두 포함하는 것으로 해석되지 않아야 하며, 그 중 일부 구성 요소들 또는 일부 단계들은 포함되지 않을 수도 있고, 또는 추가적인 구성 요소 또는 단계들을 더 포함할 수 있는 것으로 해석되어야 한다. 또한, 명세서에 기재된 "...부", "모듈" 등의 용어는 적어도 하나의 기능이나 동작을 처리하는 단위를 의미하며, 이는 하드웨어 또는 소프트웨어로 구현되거나 하드웨어와 소프트웨어의 결합으로 구현될 수 있다.

[0017] 이하, 본 발명의 대상이 되는 소프트웨어 정의 네트워크에 대해 간략하게 설명하기로 한다.

[0018] 도 1은 소프트웨어 정의 네트워크(SDN: Software Defined Network)의 기본 구조를 도시한 도면이고, 도 2는 소프트웨어 정의 네트워크에 사용되는 OpenFlow의 구조를 도시한 도면이다.

[0019] 도 1을 참조하면, 소프트웨어 정의 네트워크는 크게 데이터 평면(data plane)과 대응되는 인프라스트럭처 계층(infrastructure layer)과, 컨트롤 평면(control plane)과 대응되는 컨트롤 계층(control layer)과, 애플리케이션 계층(application layer)으로 나뉜다. 데이터 계층은 소프트웨어 정의 네트워크의 특정 인터페이스를 통해 제어를 받는 계층으로서, 데이터 흐름의 전송을 담당한다. 컨트롤 계층은 데이터의 흐름을 제어하는 계층으로서 애플리케이션과 네트워크 서비스를 통하여 데이터 흐름을 라우팅 할 것인지, 전달을 할 것인지, 거절할 것인지를 결정한다. 또한 데이터 계층의 동작들을 정리하여 API(Application Programming Interface) 형태로 애플리케이션 계층에 전달한다. 마지막으로 애플리케이션 계층은 제어 계층에서 제공한 API들을 이용하여 네트워크의 다양한 기능들을 수행 할 수 있도록 한다.

[0020] 한편, 전통적인 네트워크에서 라우터, 스위치와 같은 네트워크 장비는 트래픽 제어와 규칙을 담당한다. 그러므로 네트워크의 라우팅 정보는 스위치와 라우터에서 저장한다. 이와 같은 네트워크 구조는 네트워크가 변화할 때마다 관리자가 관련 인터넷 설비를 배치해야 한다는 문제가 있고, 데이터 센터나 그룹 네트워크 환경은 잦은 네트워크 변화로 자원을 낭비한다.

[0021] OpenFlow은 위와 같은 전통적인 네트워크의 단점을 보완하는 컨트롤러와 네트워크 장치간의 인터페이스 규격으로 사용되고 있는 기술이다. 도 2를 참조하면, OpenFlow는 제어 평면과 데이터 평면을 분리하여 네트워크를 운용할 수 있게 함으로써 네트워크 트래픽을 제어할 수 있는 기능과 전달할 수 있는 기능을 분리하며 소프트웨어를 제작하여 네트워크를 제어할 수 있도록 해준다. OpenFlow 프로토콜을 사용하면, 제어 및 데이터 평면을 하드웨어가 아닌 소프트웨어로도 구현할 수 있으며, 이 소프트웨어를 범용 서버에 설치하여 신속하게 새로운 기능을 구현할 수 있다.

[0022] OpenFlow는 프로토콜 계층 1~4까지의 헤더 정보를 하나로 조합하여 패킷(프레임)의 동작을 지정할 수 있다. 제어 평면의 프로그램을 수정하면 계층 4까지의 범위에서 사용자가 자유롭게 새로운 프로토콜을 만들 수 있고, 특정 서비스나 애플리케이션에 최적화된 네트워크를 사용자가 구현할 수도 있다. 즉, OpenFlow는 패킷을 제어하는 기능과 전달하는 기능을 분리하고 프로그래밍을 통해 네트워크를 제어하는 기술이다.

[0023] 상기에서 설명된 내용을 참조하여 본 발명의 일 실시예에 따른 악성 애플리케이션을 방지할 수 있는 소프트웨어 정의 네트워크 및 이에 포함되는 판단 장치를 상세하게 설명한다.

[0024] 도 3은 본 발명의 일 실시예에 따른 소프트웨어 정의 네트워크(SDN: Software Defined Network)의 개략적인 구조를 도시한 도면이다.

[0025] 도 3을 참조하면, 본 발명의 일 실시예에 따른 소프트웨어 정의 네트워크(300)는, 적어도 하나의 애플리케이션 계층 장치(310), 컨트롤러(320), 복수의 스위치(330) 및 판단 장치(340)를 포함한다.

[0026] 적어도 하나의 애플리케이션 계층 장치(310)는 최상층인 애플리케이션 계층 상에 위치하는 장치로서, 네트워크 관리, 컨트롤 및 모니터링 등과 같은 기능을 수행한다.

[0027] 컨트롤러(320)는 일례로, OpenFlow 인터페이스를 따르는 OF 컨트롤러일 수 있으며, 컨트롤 계층(컨트롤 평면)에 위치한다. 컨트롤러(320)는 네트워크의 모든 제어 명령, 데이터 트래픽의 전달을 수행하며, 전체 네트워크를 직접적으로 제어한다.

[0028] 여기서, 컨트롤러(320)에는 적어도 하나의 애플리케이션이 설치되어 특정 동작을 수행한다. 또한, 컨트롤러(320)에는 애플리케이션을 사용하기 위한 API(Application Programming Interface)가 설치되어 다른 모듈 또는

애플리케이션으로 접근할 수 있다. 특히, 애플리케이션 계층 장치(310)는 컨트롤러(320)에 설치된 애플리케이션을 사용하기 위해 API 요청을 컨트롤러(320)로 전송한다.

[0029] 복수의 스위치(330) 각각은 일례로, OpenFlow 인터페이스를 따르는 OF 스위치일 수 있으며, 인프라스트럭처 계층(데이터 평면)에 위치하며, 대응되는 외부 네트워크와 연결된다.

[0030] 즉, 컨트롤러(320)는 복수의 스위치(330) 각각에 명령을 전송하고, 각각의 스위치(330)는 수신된 명령에 따라 패킷을 목적지로 전송하거나 수정, 폐기하는 등의 처리를 한다. OpenFlow 프로토콜을 이용하여, 컨트롤러(320)는 패킷의 포워딩 방법이나 VLAN 우선순위 값 등을 스위치(330)에 전달하여 수행되도록 하며, 스위치(330)는 장애정보와 사전에 등록된 플로우 엔트리가 없는 패킷에 대한 정보를 컨트롤러에 문의하고 그 결정을 받아 처리한다.

[0031] 한편, 컨트롤러(320)에서 설치되는 애플리케이션은 정상 애플리케이션일 수 있고 악성 애플리케이션일 수도 있다. 이 때, 악성 애플리케이션은 컨트롤러(320)에 잘못된 규칙들(rules) 또는 악성 규칙들을 설치하여 컨트롤러(320)의 오작동을 유발한다. 즉, 컨트롤러(320)는 컨트롤러(320)의 핵심 모듈에 액세스할 수 애플리케이션 계층을 위한 API를 제공하는데, 악성 애플리케이션은 컨트롤러(320)의 API의 악용을 유발할 수 있다.

[0032] 판단 장치(340)는 상기에서 설명한 문제점을 해결하기 위한 것으로서, 애플리케이션이 정상 애플리케이션인지 악성 애플리케이션인지 여부를 판단하며, 악성 애플리케이션인 경우 악성 애플리케이션의 공격을 방지하는 기능을 수행한다.

[0033] 즉, 판단 장치(340)는 애플리케이션 계층과 컨트롤 계층 사이에 존재하며, 컨트롤러(320)에 설치된 애플리케이션을 사용하기 위한 API 요청을 애플리케이션 계층으로부터 수신하고, 수신된 API 요청이 정상 애플리케이션을 위한 API 요청인지 악성 애플리케이션을 위한 API 요청인지 여부를 판단한다. 만약, 수신된 API 요청이 정상 애플리케이션을 위한 API 요청인 경우, 판단부(340)는 수신된 API 요청을 컨트롤러(320)로 전송한다. 반대로, 수신된 API 요청이 악성 애플리케이션을 위한 API 요청인 경우, 판단부(340)는 수신된 API 요청을 컨트롤러(320)로 전송하지 않는다. 따라서, 악성 애플리케이션의 실행이 방지된다.

[0034] 보다 상세하게, 판단 장치(340)는 통신부(341) 및 정책 엔진부(Policy Engine)(342)를 포함한다.

[0035] 통신부(341)은 애플리케이션 계층에서 전송된 API 요청(request API)을 수신하는 기능(catching API)을 수행하며, 이를 정책 엔진부(324)로 전달한다.

[0036] 정책 엔진부(342)는 정상 애플리케이션에 대한 API 요청인지 여부를 판단하는 정책 정보를 이용하여 상기 수신된 API 요청이 정상 애플리케이션을 위한 API 요청인지 악성 애플리케이션을 위한 API 요청인지 여부를 판단한다. 또한, 정책 엔진부(342)는 정책 규칙 데이터베이스를 포함하며, 이는 컨트롤러 구성(controller configuration), 등록된 모듈들의 리스트(list of registered modules) 등 컨트롤러(320)의 불변량(invariants)를 저장한다.

[0037] 만약, 정책 엔진부(342)가 수신된 API 요청이 정상 애플리케이션을 위한 API 요청인 것으로 판단하는 경우, 통신부(341)는 수신된 API 요청을 컨트롤러(320)로 전송한다. 반대로, 정책 엔진부(342)가 수신된 API 요청이 악성 애플리케이션을 위한 API 요청인 것으로 판단하는 경우, 통신부(341)는 수신된 API 요청을 컨트롤러(320)로 전송하지 않는다.

[0038] 여기서, 정책 정보는 제1 정책 정보, 제2 정책 정보, 제3 정책 정보 및 제4 정책 정보를 포함하며, 정책 엔진부(342)는 수신된 API 요청에 포함되는 정보와 정책 정보들을 비교하여 애플리케이션의 정상 여부를 판단한다. 즉, 정책 엔진부(342)는 수신된 API 요청에 포함되는 정보가 정책 정보들 중 적어도 하나, 바람직하게는 정책 정보들을 모두를 통과하는 경우 수신된 API 요청이 정상 애플리케이션을 위한 API 요청인 것으로 판단한다.

[0039] 본 발명의 일 실시예에 따르면, 제1 정책 정보는 API에 접근 가능한 적어도 하나의 애플리케이션의 식별정보일 수 있다. 이 경우, 정책 엔진부(342)는 수신된 API 요청에 포함된 애플리케이션의 식별정보가 제1 정책 정보에 포함된 적어도 하나의 애플리케이션의 식별정보 중 하나인 경우, 수신된 API 요청이 상기 정상 애플리케이션을 위한 API 요청인 것으로 판단한다.

[0040] 즉, 제1 정책 정보는 액세스 정책 규칙(Access Policy Rules)으로서, 이에 대해 상세하게 설명하면 다음과 같다.

[0041] 액세스 정책은 컨트롤러(320) 코드의 정적 분석을 수행하고, 컨트롤러(320)에 설치된 API에 애플리케이션이 접

근할 수 있는지를 확인하는 것으로 정의된다. 정책 규칙 데이터베이스는 "permission.csv" 파일을 저장하며, "permission.csv" 내에서 각각의 API 및 모듈에 대해 접근 정책을 읽는다. 정책 엔진부(320)는 컨트롤러 스타트업에서 "permission.csv" 파일을 읽고 정책 규칙 데이터베이스에 이를 저장한다. 정책 엔진부(342)는 런타임에 호출 스택을 덤프(dump)하며, 모듈이 컨트롤러(320)의 API를 호출하는지를 식별한다. 그 후, 정책 엔진부(342)는 컨트롤러(320)의 API에 대한 권한 집합을 식별하기 위해 정책 규칙 데이터베이스를 검색한다. 호출 모듈에 대한 권한 집합이 읽기 또는 쓰기 등과 같은 유효한 값을 포함하는 경우 다음 과정이 수행된다. 만약, 접근 정책을 설정하지 않는 경우, 컨트롤러(320)의 API에 대한 실행이 필요없어 지고, API 후크(hook)는 실패 응답을 반환한다. 액세스 권한에 대한 변경 사항은 정책 규칙 데이터베이스에 의해 동적으로의 처리될 수 있다. 예를 들어, "permission.csv" 파일 내에 어떠한 변화가 존재하는 경우, 정책 엔진부(342)는 정책 규칙 데이터베이스를 업데이트하고, 컨트롤러(320)의 API에 대한 추가 접근이 처리된다.

[0042] 또한, 본 발명의 일 실시예에 따르면, 제2 정책 정보는 API로 전달될 수 있는 불변(invariant) 데이터의 정보일 수 있다. 이 경우, 정책 엔진부(342)는 수신된 API 요청에 포함된 불변 데이터 정보가 제2 정책 정보에 포함된 불변 데이터의 정보에 포함되는 경우, 수신된 API 요청이 정상 애플리케이션을 위한 API 요청인 것으로 판단할 수 있다.

[0043] 즉, 제2 정책 정보는 구문 정책 규칙(Access Policy Rules)으로서, 이에 대해 상세하게 설명하면 다음과 같다.

[0044] 구문 정책은 파라미터에 대한 불변 데이터의 사용을 정의한다. 정책 엔진부(342)는 컨트롤러(320) 코드 내의 불변 데이터를 확인하기 위해 정적 분석 도구를 사용한다. 정책 엔진부(342)는 정책 규칙 데이터베이스의 값에 대해서 컨트롤러(320)의 API에서 전달된 불변 데이터를 검사한다. 악성 값이 발견되는 경우 컨트롤러(320)의 API에 대한 액세스가 차단된다.

[0045] 또한, 본 발명의 일 실시예에 따르면, 제3 정책 정보는 기 설정된 범위에서 변화하는 동적 데이터의 정보일 수 있다. 이 경우, 정책 엔진부(320)는 수신된 API 요청에 포함된 동적 데이터가 제3 정책 정보에 포함된 동적 데이터의 정보에 포함되는 경우, 수신된 API 요청이 정상 애플리케이션을 위한 API 요청인 것으로 판단할 수 있다.

[0046] 즉, 제3 정책 정보는 시맨틱 정책 규칙(Semantic Policy Rules)으로서, 시멘틱 정책은 동적 데이터에 대해 범위 내에서의 변화가 발생하는지 여부로 정의된다. 일례로, IP 주소, 포트 및 구성 데이터가 동적 데이터의 예이다.

[0047] 또한, 본 발명의 일 실시예에 따르면, 제4 정책 정보는 2개의 모듈 사이의 통신의 흐름 정보일 수 있다. 이 때, 모듈은 인프라스트럭처 계층에 존재하는 모듈이다. 이 경우, 정책 엔진부(342)는 수신된 API 요청에 포함된 통신의 흐름 정보가 제4 정책 정보에 포함된 통신의 흐름 정보에 포함되는 경우, 수신된 API 요청이 정상 애플리케이션을 위한 API 요청인 것으로 판단할 수 있다.

[0048] 즉, 제4 정책 정보는 통신 정책 규칙(Communication Policy Rules)으로서, 이에 대해 상세하게 설명하면 다음과 같다.

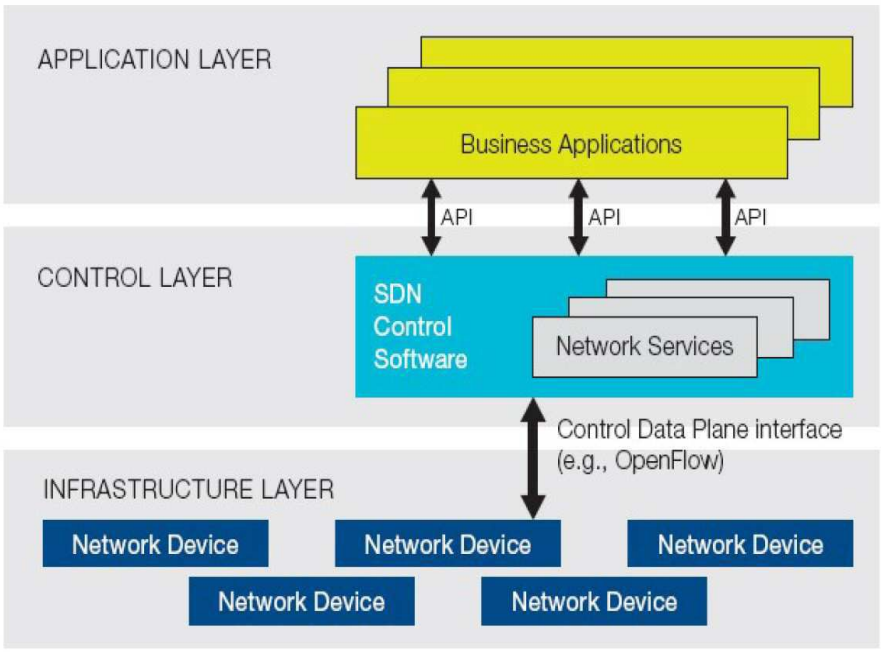
[0049] 통신 정책은 요청들의 실행의 흐름을 정의한다. 통신 정책은 두 개의 모듈 사이의 통신 관계 사이의 프로토콜의 상태를 확인한다. 일례로서, 호스트는 현재 포트의 적절한 종료 없이 다른 스위치 포트로 이동하지 않아야 한다. 이 정책은 두 개의 모듈 또는 인터페이스 사이의 통신에서의 그러한 위반을 감지한다. 그리고, 정책 엔진은 검증을 위한 통신의 상태를 유지한다.

[0050] 요컨대, 본 발명의 일 실시예에 따른 소프트웨어 정의 네트워크(300) 및 이에 포함되는 판단 장치(340)는 컨트롤러(320)의 API를 오용으로부터 보호하며, 악성 애플리케이션의 공격으로부터 컨트롤러(320)를 보호하는 장점이 있다.

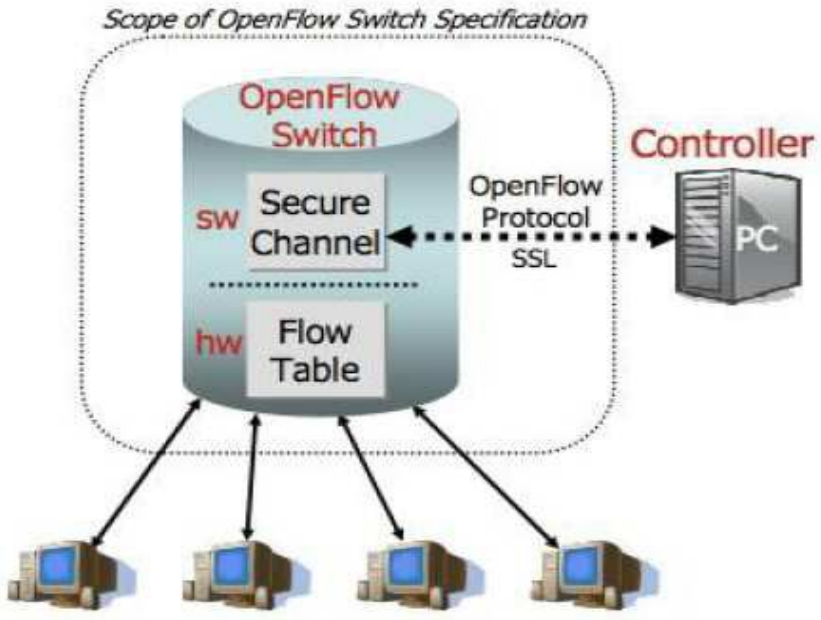
[0051] 이상과 같이 본 발명에서는 구체적인 구성 요소 등과 같은 특정 사항들과 한정된 실시예 및 도면에 의해 설명되었으나 이는 본 발명의 전반적인 이해를 돕기 위해서 제공된 것일 뿐, 본 발명은 상기의 실시예에 한정되는 것은 아니며, 본 발명이 속하는 분야에서 통상적인 지식을 가진 자라면 이러한 기재로부터 다양한 수정 및 변형이 가능하다. 따라서, 본 발명의 사상은 설명된 실시예에 국한되어 정해져서는 아니되며, 후술하는 특허청구범위뿐 아니라 이 특허청구범위와 균등하거나 등가적 변형이 있는 모든 것들은 본 발명 사상의 범주에 속한다고 할 것이다.

도면

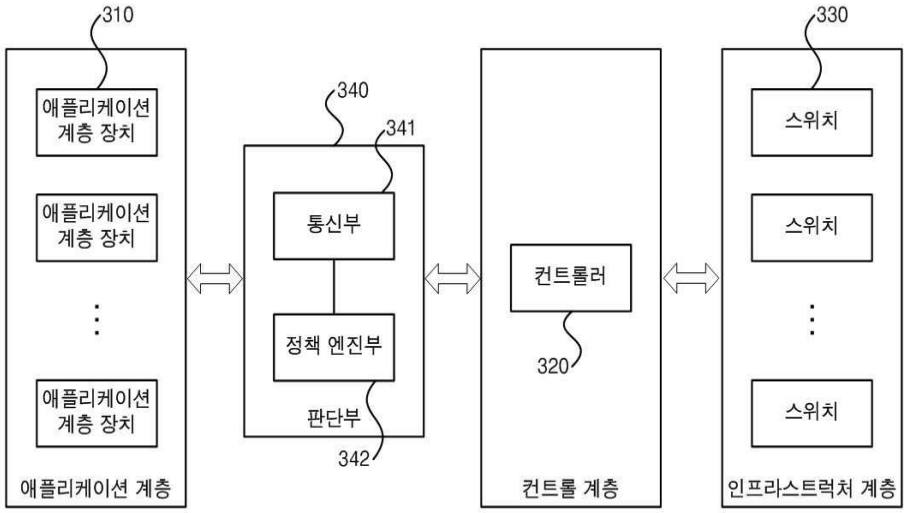
도면1


APPLICATION LAYER
Business Applications
API
API
API
CONTROL LAYER
SDN
Control
Software
Network Services
Control Data Plane interface
(e.g., OpenFlow)
INFRASTRUCTURE LAYER
Network Device
Network Device
Network Device
Network Device
Network Device

도면2


Scope of OpenFlow Switch Specification
OpenFlow
Switch
sw
Secure
Channel
hw
Flow
Table
OpenFlow
Protocol
SSL
Controller
PC

도면3


310
애플리케이션 계층 장치
애플리케이션 계층 장치
애플리케이션 계층 장치
애플리케이션 계층
340
341
통신부
정책 엔진부
판단부
342
컨트롤러
320
컨트롤 계층
330
스위치
스위치
스위치
인프라스트럭처 계층